UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/602,916	06/25/2003	Patrick Knebel	10971390-3	1768

22879 7590 01/25/2006

HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER	
MEONSKE, TONIA L	

ART UNIT	PAPER NUMBER
2181	

DATE MAILED: 01/25/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/602,916	KNEBEL ET AL.	
	Examiner	Art Unit	
	Tonia L Meonske	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *June 25th 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-17* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-5,7-11 and 13-17* is/are rejected.

7)☒ Claim(s) *6 and 12* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *25 June 2005* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *6/25/2003*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This is a continuation of US Application 09/496,845, now US Patent 6,618,801.

2. Claims 1-17 have been examined.

Double Patenting

3. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., In re Berg, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); In re Goodman, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); In re Longi, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); In re Van Ornum, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); In re Vogel, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and In re Thorington, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

4. A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

5. Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

6. Claim 1 is rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claim 7 of U.S. Patent No. 6,618,801. Although the conflicting claims are not identical, they are not patentably distinct from each other because claim 7 of patent # 6,618,801 contains every element of claim 1 of the instant application and as such anticipates claim 1 of the instant application.

> a. "A later patent claim is not patentably distinct from an earlier patent claim if the later claim is obvious over, or **anticipated by**, the earlier claim. In re Longi, 759 F.2d at 896, 225 USPQ at 651 (affirming a holding of obviousness-type double patenting because the claims at issue were obvious over claims in four prior art patents); In re Berg, 140 F.3d at 1437, 46 USPQ2d at 1233 (Fed. Cir. 1998) (affirming a holding of obviousness-type double patenting where a patent application claim to a genus is anticipated by a patent claim to a species within that genus). " ELI LILLY AND COMPANY v BARR LABORATORIES, INC., United States Court of Appeals for the Federal Circuit, ON PETITION FOR REHEARING EN BANC (DECIDED: May 30, 2001).

7. Claims 7 and 16 are rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claim 1 of U.S. Patent No. 6,618,801 (herein after referred to as '801) in view of Hull, US Patent 5,922,065 (herein after referred to as Hull).

8. Referring to claim 7, claim 1 of '801 has taught each and every limitation of claim 7, except for "converting the one or more microinstructions into a bundle". However, claim 7 has claimed "formatting the microinstructions into bundles". The "formatting" and the "converting"

produce the same result, so they are functionally equivalent. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to have claim 1 "formatting" instead of "converting" since they are functionally equivalent. Furthermore, claim 1 has not taught "dispatching additional information to the execution engine, wherein the additional information is contained in bits of an instruction bundle otherwise not required for execution of the instruction bundle." However, Hull has taught dispatching additional information to the execution engine, wherein the additional information is contained in bits of an instruction bundle otherwise not required for execution of the instruction bundle (Hull, abstract, Figures 3 and 4, column 2, lines 21-29; column 4, lines 20-32, Hull issues a template field with a bundle of instructions. The template field is not required for execution of the instructions. The template field maps the instructions in the slots to execution unit types in order to increase processor efficiency during execution time (Hull, column 2, lines 5-30).). Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to have claim 1 of '801, include dispatching additional information (i.e. the templates of Hull) to the execution engine, as claimed in claim 7, wherein the additional information is contained in bits of the bundle otherwise not required for emulation of the macroinstruction, as taught by Hull, for the desirable purpose of increasing processor efficiency (Hull, column 2, lines 5-30).

9. Referring to claim 16, claim 1 of '801 has taught each and every limitation of claim 16, except for "transferring, by the emulation engine, additional information to the execution engine, wherein the additional information, including control information from the emulation front end that is sent using a memory, floating-point, integer ("MFI") template, wherein the MFI template specifies that the bundle includes a memory instruction in a first syllable, a floating point

instruction in a second syllable, and an integer instruction in a third syllable." However, Hull has taught "transferring, by the emulation engine, additional information to the execution engine, wherein the additional information, including control information from the emulation front end that is sent using a memory, floating-point, integer ("MFI") template, wherein the MFI template specifies that the bundle includes a memory instruction in a first syllable, a floating point instruction in a second syllable, and an integer instruction in a third syllable" (Hull, figures 2-4, template 6 of figure 4, abstract, column 2, lines 21-29; column 4, lines 20-32, Hull issues a template field with a bundle of instructions. The template field maps the instructions in the slots to execution unit types in order to increase processor efficiency during execution time (Hull, column 2, lines 5-30).). Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to have claim 1 of '801 include "transferring, by the emulation engine, additional information to the execution engine, wherein the additional information, including control information from the emulation front end that is sent using a memory, floating-point, integer ("MFI") template, wherein the MFI template specifies that the bundle includes a memory instruction in a first syllable, a floating point instruction in a second syllable, and an integer instruction in a third syllable", as taught by Hull, for the desirable purpose of increasing processor efficiency (Hull, column 2, lines 5-30).

10. Claims 13 is rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claim 1 of U.S. Patent No. 6,618,801 (herein after referred to as '801) in view of Hull, US Patent 5,922,065 (herein after referred to as Hull) and Nemirovsky et al., US Patent 6,105,125 (herein after referred to as Nemirovsky). Claim 1 of '801 has each and every limitation of claim 13, except for "transferring, by the emulation engine, additional information

to the execution engine, wherein the additional information includes an immediate from an emulation front end that is sent by using a memory, long-immediate, integer ("MLI") template that is interpreted by the execution engine differently, depending upon whether the execution engine is operating in native mode or emulation mode. However, Nemirovsky has taught dispatching an immediate, which is merely a part of the claimed additional information, from the emulation front end (Nemirovsky, column 3, lines 48-52). It would be obvious to one of ordinary skill in the art at the time of the invention to include an immediate value to the execution engine of claim 13 so that the execution unit would be able to correctly execute the instruction. If an instruction had an immediate value associated with it, that value would have been required for the proper result of the instruction. Allowing the immediate to have its own syllable, or template, would have made it easier for the decoder to recognize the immediate value and would have reduced the time required for decoding. Therefore, one of ordinary skill in the art at the time of the invention would have included this extra information in an immediate template to reduce decoding time and to get the proper results of instruction execution.

11. The combination of claim 1 and Nemirovsky have not specifically taught wherein the immediate template is sent by using an MLI template that is interpreted by the execution engine differently, depending upon whether the execution engine is operating in a native mode or an emulation mode. Davidson has taught an immediate template, sent by using an MLI template, that is interpreted by the execution engine differently, depending upon what mode the execution engine is operating in (Davidson, column 26, lines 14-28; The different interpreters interpret the same template in different ways, with each pass of the template using a different mode.). By allowing the same template to have been interpreted in different ways for different modes, the

system would have understood several different languages, one for each pass. It would have been obvious to one of ordinary skill in the art at the time of the invention to have the same template interpreted by two different modes, one being a native mode and one being an emulation mode, such that the template would have been interpreted in different ways. Emulation mode was used so that one processor could understand data or instructions in a different way than its normal execution so that the processor could execute a different type of architecture than originally designed. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to interpret the template of claim 1 differently for different modes, so that the system could have used and emulated several different computer languages, and therefore take advantage of previously written code that existed in numerous different languages.

Claim Rejections - 35 USC § 103

12. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

13. Claims 1-3 are rejected under 35 U.S.C. 103(a) as being unpatentable over Panwar et al., US Patent 5,875,316 (herein referred to as Panwar), in view of Hull et al., US Patent 5,922,065 (herein referred to as Hull).

14. Referring to claim 1, Panwar has taught a method for implementing two types of architectures on a chip, comprising:

a. receiving an instruction from a fetch engine (Panwar, column 5, lines 59-60, figure 2, reference number 202),

b. determining whether the instruction is a macroinstruction or a microinstruction (Panwar, column 7, line 61-column 8 line 2, figures 3 and 6),

c. if the instruction is a macroinstruction, sending the macroinstruction to an emulation engine (Panwar, column 7, line 67-column 8 line 2),

d. decomposing the macroinstruction into one or more microinstructions (Panwar, column 10, lines 32-34),

e. formatting, by a bundler, the microinstructions into bundles as preferred by the native microarchitecture (Panwar, column 10, lines 34-36),

f. dispatching a bundle in parallel to an execution engine (Panwar, column 10, lines 36-38),

g. if the instruction is microinstruction, dispatching the microinstruction to the execution engine (Panwar, column 7, lines 65-67 and column 8, lines 15-17).

15. Panwar has taught dispatching additional information to the execution engine (Panwar, column 10, line 55-column 11, line 10).

16. Panwar has not specifically taught dispatching additional information to the execution engine, wherein the additional information is contained in bits of the bundle otherwise not required for emulation of the macroinstruction. However, Hull has taught dispatching additional information to the execution engine, wherein the additional information is contained in bits of an instruction bundle otherwise not required for execution of the instruction bundle (Hull, abstract, Figures 3 and 4, column 2, lines 21-29; column 4, lines 20-32, Hull issues a template field with a bundle of instructions. The template field is not required for execution of the instructions. The template field maps the instructions in the slots to execution unit types in order to increase

processor efficiency during execution time (Hull, column 2, lines 5-30).). Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to have the method of Panwar, include dispatching additional information (i.e. the templates of Hull) to the execution engine, wherein the additional information is contained in bits of the bundle otherwise not required for emulation of the macroinstruction, as taught by Hull, for the desirable purpose of increasing processor efficiency (Hull, column 2, lines 5-30).

17. Referring to claim 2, Panwar in combination with Hull have taught the method of claim 1, as described above, and further comprising,

a. selecting either the microinstruction from the fetch engine or the bundle from the emulation engine, by using a multiplexer (Panwar, column 7, line 61-column 8, line 2), and

b. dispatching the selected instruction to the execution engine (Panwar, column 8, lines 15-17; column 10, lines 34-37).

18. Referring to claim 3, Panwar in combination with Hull have taught the method according to clam 1, as described above, and wherein the additional information includes control information from the emulation front end (Panwar, column 10, lines 57-60) that is sent using a memory, floating-point, integer ("MFI") template, wherein the MFI template specifies that the bundle includes a memory instruction in a first syllable, a floating point instruction in a second syllable, and an integer instruction in a third syllable (Hull, figures 2-4, template 6 of figure 4).

19. Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over Panwar et al., US Patent 5,875,316 (herein referred to as Panwar), in view of Hull et al., US Patent 5,922,065 (herein referred to as Hull), as applied to claim 1 above, and further in view of Nemirovsky et al.,

US Patent 6,105,125 (herein referred to as Nemirovsky), and Davidson et al., US Patent 5,613,117 (herein referred to as Davidson).

20. Referring to claim 4, Panwar and Hull have taught the method according to clam 1, as described above. They have not specifically taught wherein the additional information includes an immediate from an emulation front end that is sent by using a memory, long-immediate, integer ("MLI") template that is interpreted by the execution engine differently, depending upon whether the execution engine is operating in native mode or emulation mode. However, Nemirovsky has taught dispatching an immediate, which is merely a part of the claimed additional information, from the emulation front end (Nemirovsky, column 3, lines 48-52). It would be obvious to one of ordinary skill in the art at the time of the invention to include an immediate value to the execution engine so that the execution unit would be able to correctly execute the instruction. If an instruction had an immediate value associated with it, that value would have been required for the proper result of the instruction. Allowing the immediate to have its own syllable, or template, would have made it easier for the decoder to recognize the immediate value and would have reduced the time required for decoding. Therefore, one of ordinary skill in the art at the time of the invention would have included this extra information in an immediate template to reduce decoding time and to get the proper results of instruction execution.

21. The combination of Panwar, Hull and Nemirovsky have not specifically taught wherein the immediate template is sent by using an MLI template that is interpreted by the execution engine differently, depending upon whether the execution engine is operating in a native mode or an emulation mode. Davidson has taught an immediate template, sent by using an MLI template,

that is interpreted by the execution engine differently, depending upon what mode the execution engine is operating in (Davidson, column 26, lines 14-28; The different interpreters interpret the same template in different ways, with each pass of the template using a different mode.). By allowing the same template to have been interpreted in different ways for different modes, the system would have understood several different languages, one for each pass. It would have been obvious to one of ordinary skill in the art at the time of the invention to have the same template interpreted by two different modes, one being a native mode and one being an emulation mode, such that the template would have been interpreted in different ways. Emulation mode was used so that one processor could understand data or instructions in a different way than its normal execution so that the processor could execute a different type of architecture than originally designed. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to interpret the template differently for different modes, so that the system could have used and emulated several different computer languages, and therefore take advantage of previously written code that existed in numerous different languages.

22. Claims 5, 7, 8, 10, and 11 are rejected under 35 U.S.C. 103(a) as being unpatentable over Panwar et al., US Patent 5,875,316 (herein referred to as Panwar), in view of Hull et al., US Patent 5,922,065 (herein referred to as Hull) and Eickemeyer et al., US Patent 5,459,844 (herein referred to as Eickemeyer).

23. Referring to claim 5, Panwar have taught the method according to clam 1, as described above. Panwar has not taught wherein the bundler

b. receives at least one sequence of instructions ("XUOPs"),

c. determines how many XUOPs are received, and

d. when more than one XUOP is received, determines whether the XUOPs must be issued in parallel.

24. Eickemeyer has taught a bundler that

a. receives at least one sequence of instructions ("XUOPs") (Eickemeyer, column 18 line 64-column 19, line 4; column 2, line 49-column 3, line 6; column 10, lines 30-34),

b. determines how many XUOPs are received (Eickemeyer, column 9, lines 26-50, column 10, lines 15-30, Figures 5a and 5b, The compounding unit determines how many XUOP's are received based on the tags and/or instruction boundaries in the instruction stream.), and

c. when more than one XUOP is received, determines whether the XUOPs must be issued in parallel (Eickemeyer, column 18, line 64-column 19, line 4; column 2, line 49-column 3, line 6), for the purpose of deciding before execution time which instructions are to executed in parallel, thereby accelerating instruction execution. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention as made to have the bundler of Panwar "receive at least one sequence of instructions ("XUOPs"), determine how many XUOPs are received, and when more than one XUOP is received, determine whether the XUOPs must be issued in parallel", as taught by Eickemeyer, for the desirable purpose of accelerating instruction execution.

25. Referring to claim 7, Panwar has taught a method for implementing two architectures on a chip, comprising,

a. decoding a macroinstruction into one or more microinstructions, through the use of an emulation engine (Panwar, abstract, column 7, line 67-column 8, line 35),

b. formatting the microinstructions into bundles, by use of a bundler, as preferred by the native microarchitecture (Panwar, abstract, column 7, line 67-column 8, line 35; column 4, lines 26-42, Figure 5, elements 203, 500A, 500B, and 500C), and

c. dispatching the bundle to an execution engine (Panwar, column 10, lines 36-38).

26. Panwar has not taught wherein the bundler

a. receives at least one sequence of instructions (an "XUOP"),

b. determines how many of the at least one XUOP are received, and

c. when more than one XUOP is received, determines whether the XUOPs must be issued in parallel.

27. However Eickemeyer has taught a bundler that

a. receives at least one sequence of instructions (an "XUOPs") (Eickemeyer, column 18, line 64-column 19, line 4; column 2, line 49-column 3, line 6; column 10, lines 30-34),

b. determines how many of the at least one XUOP are received (Eickemeyer, column 9, lines 26-50; column 10, lines 15-30, Figures 5a and 5b, The compounding unit determines how many XUOP's are received based on the tags and/or instruction boundaries in the instruction stream.), and

c. when more than one XUOP is received, determines whether the XUOPs must be issued in parallel (Eickemeyer, column 18, line 64-column 19, line 4; column 2 line 49-column 3, line 6), for the purpose of deciding before execution time which instructions should be executed in parallel, thereby accelerating instruction execution. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention as

made to have the bundler of Panwar "receive at least one sequence of instructions ("XUOPs"), determine how many XUOPs are received, and when more than one XUOP is received, determine whether the XUOPs must be issued in parallel", as taught by Eickemeyer, for the desirable purpose of accelerating instruction execution.

28. Furthermore Panwar has taught dispatching additional information to the execution engine (Panwar, column 10, line 55-column 11, line 10). Panwar has not specifically taught dispatching additional information to the execution engine, wherein the additional information is contained in bits of the bundle otherwise not required for emulation of the macroinstruction. However, Hull has taught dispatching additional information to the execution engine, wherein the additional information is contained in bits of an instruction bundle otherwise not required for execution of the instruction bundle (Hull, abstract, Figures 3 and 4, column 2, lines 21-29; column 4, lines 20-32, Hull issues a template field with a bundle of instructions. The template field is not required for execution of the instructions. The template field maps the instructions in the slots to execution unit types in order to increase processor efficiency during execution time (Hull, column 2, lines 5-30).). Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to have the method of Panwar, include dispatching additional information (i.e. the templates of Hull) to the execution engine, wherein the additional information is contained in bits of the bundle otherwise not required for emulation of the macroinstruction, as taught by Hull, for the desirable purpose of increasing processor efficiency (Hull, column 2, lines 5-30).

29. Claim 8 has nothing over claim 4, and is therefore rejected for the same reasons as set forth in claim 4.

30. Claim 10 claims nothing over claim 3, and is therefore rejected for the same reasons as set forth in claim 3.

31. Referring to claim 11, Panwar, Eickemeyer, and Hull have taught the method of claim 7, as described above, and wherein the emulation engine delivers a pre-decode bit to the execution engine along with the bundle (Eickemeyer, column 18 line 64-column 19, line 4; column 2 line 57-column 3 line 6).

32. Claim 9 is rejected under 35 U.S.C. 103(a) as being unpatentable over Panwar et al., US Patent 5,875,316 (herein referred to as Panwar), in view of Hull et al., US Patent 5,922,065 (herein referred to as Hull) and Eickemeyer et al., US Patent 5,459,844 (herein referred to as Eickemeyer), and Davidson et al., US Patent 5,613,117 (herein referred to as Davidson).

33. Referring to claim 9, Panwar, Eickemeyer and Hull have taught the method of claim 8, as described above, and wherein, the MLI template specifies that a third syllable of the bundle contains an integer instruction that operates on an immediate located in second and third syllables of the bundle (Hull, Figures 2-4, Template 2 of Figure 4, column 5 lines 39-49; The 3rd syllable is restricted to certain instructions that use the long immediate value in the 2nd syllable, ie. movl is the move long instruction. The movl instruction would have taken the 2nd and 3rd syllables, or slot1 and slot 2.), and the MLI template specifies that the third syllable of the bundle contains an integer instruction that operates on an immediate located entirely within the second syllable (Hull, Figures 2-4, Template 1 of Figure 4; Since the immediate is not long, only one syllable is taken up by the immediate, the syllable with the integer, the other syllables are used for other instructions).

34. The combination of Panwar, Eickemeyer and Hull has not taught wherein the template is interpreted by the execution engine differently, depending upon whether the execution engine is operating in native mode or emulation mode. Davidson has taught wherein the template is interpreted by the execution engine differently, depending upon whether the execution engine is operating in native mode or emulation mode (Davidson, column 26 lines 14-28; The different interpreters interpret the same template in different ways, with each pass of the template using a different mode.). By allowing the same template to be interpreted in different ways for different modes, the system can understand several different languages, one for each pass. It would have been obvious to one of ordinary skill in the art at the time of the invention that the same template being interpreted by two different modes, one being a native mode and one being a emulation mode, that the template would be interpreted in different ways. Emulation mode is used so that one processor can understand data or instructions in a different way than its normal execution so that it can execute a different type of architecture. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to interpret the template differently for different modes, so that the system can use several different computer languages, and therefore, can take advantage of previously written code that is in different languages.

Allowable Subject Matter

35. Claims 6 and 12 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

36. Claims 13-17 have been rejected under nonstatutory double patenting, but they have not been rejected based upon prior art. Claims 13-17 would be allowable if the rejections were overcome by a timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d).

37. The following is a statement of reasons for the indication of allowable subject matter:

a. Panwar has taught decoding macroinstructions into microinstructions, and then converting microinstructions into bundles to be dispatched and issued to an execution unit. Panwar has not taught, separately or in combination, the specific rules that are used to determine whether the XUOP's must be issued in parallel, as in claim 12, and similarly in claims 6 and 13-17.

b. Hull has taught grouping instructions into bundles containing fixed instructions slots and mapping instruction slots to the execution units via encoded templates. Hull has not taught, separately or in combination, the specific rules that are used to determine whether XUOP's must be issued in parallel, as in claim 12, and similarly in claims 6 and 13-17.

c. Eickemeyer has taught tagging instructions in main memory. At issue time the tags are examined for parallel issue. Those tagged for parallel issue are sent to different ones of the functional units depending on their opcode fields. Eickemeyer has not taught, separately or in combination, the specific rules that are used to determine whether XUOP's must be issued in parallel, as in claim 12, and similarly in claims 6 and 13-17.

d. Nemirovsky has taught using predecoded instruction prefixes on instructions to increase instruction flexibility by using a generic instruction for many instructions. Eickemeyer has not taught, separately or in combination, any rules that are used to

determine whether XUOP's must be issued in parallel, as in claim 12, and similarly in claims 6 and 13-17.

e. Davidson has taught an immediate template, sent by using an MLI template, that is interpreted by the execution engine differently, depending upon what mode the execution engine is operating in, i.e. The different interpreters interpret the same template in different ways, with each pass of the template using a different mode. Davidson has not taught, separately or in combination, rules that are used to determine whether XUOP's must be issued in parallel, as in claim 12, and similarly in claims 6 and 13-17.

f. The claimed rules that are used to determine whether the XUOP's must be issued in parallel are not found in the prior art in combination with the remaining claim limitations. Furthermore, the claimed rules in combination with the remaining limitations in the claims would not have been obvious to one of ordinary skill in the art at the time the invention was made.

Conclusion

38. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tonia L Meonske whose telephone number is (571) 272-4170. The examiner can normally be reached on Monday-Friday, 8-4:30.

39. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kim Huynh can be reached on (571) 272-4147. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

40. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

tlm

1/18/06

HENRY W. H. TSAI
PRIMARY EXAMINER